

Figure 1

′ء



,0Z

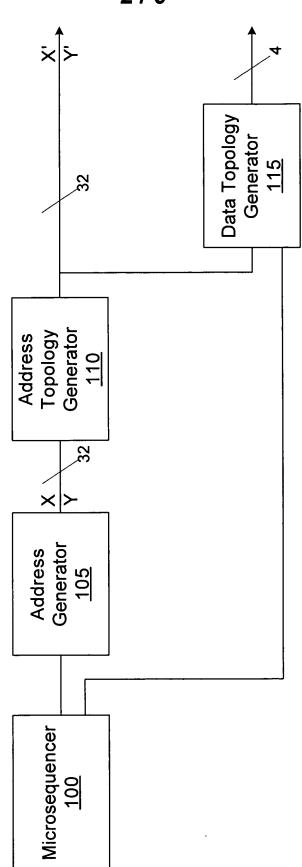
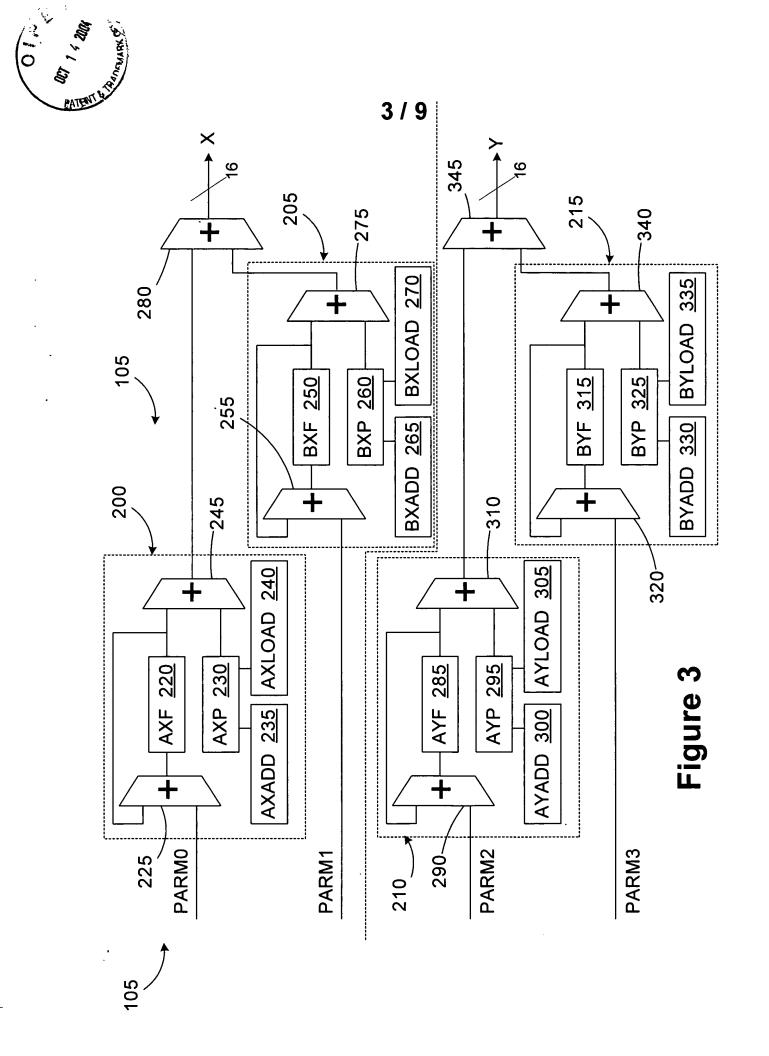
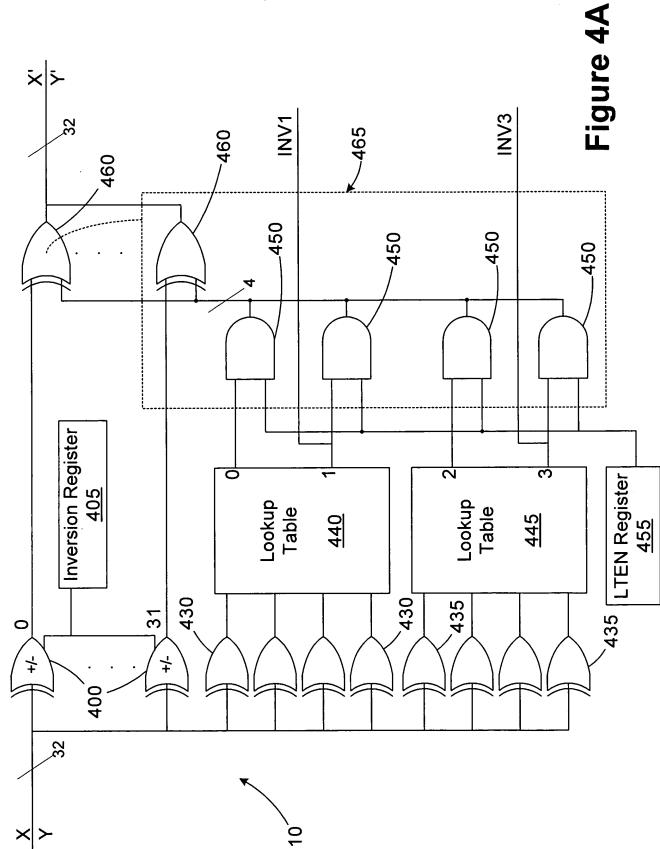


Figure 2







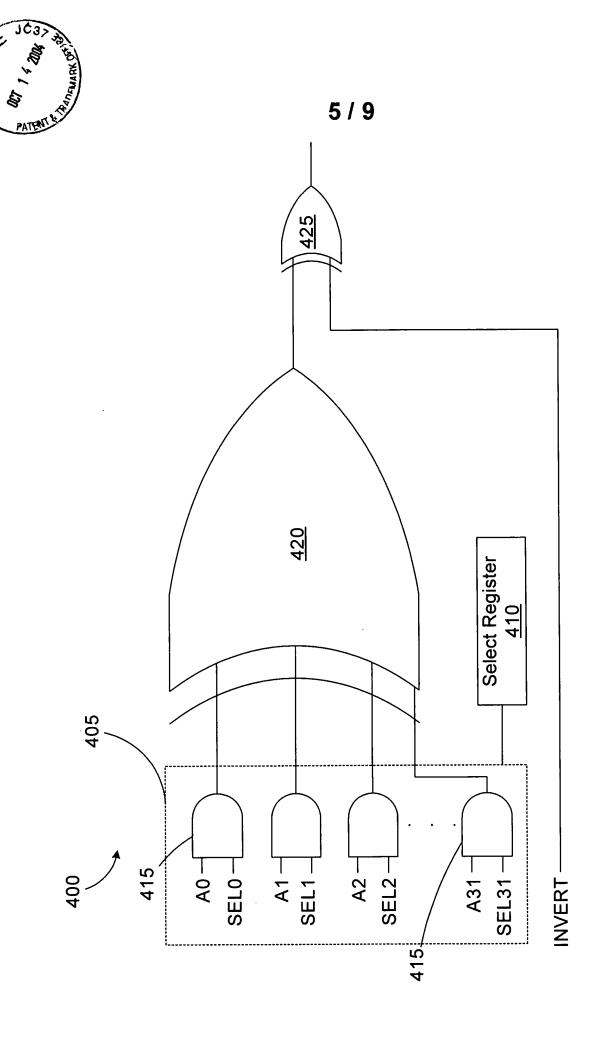
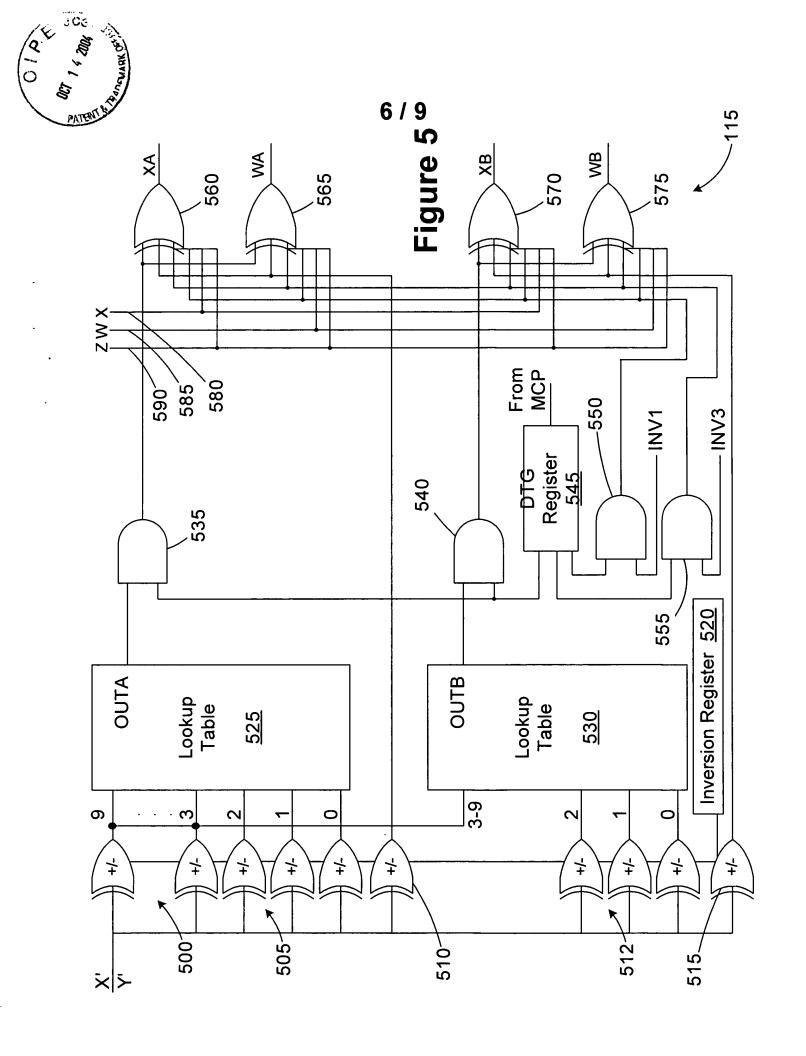
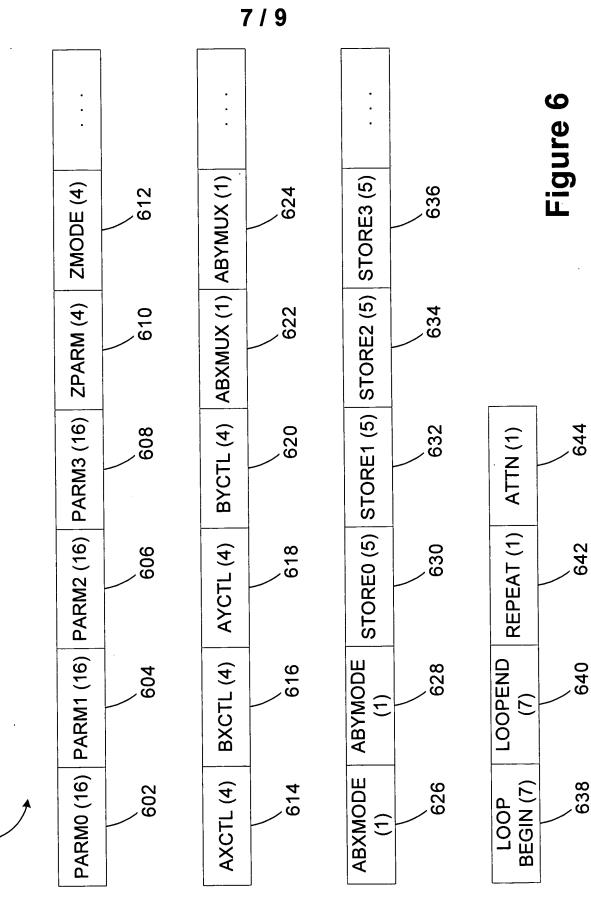


Figure 4B





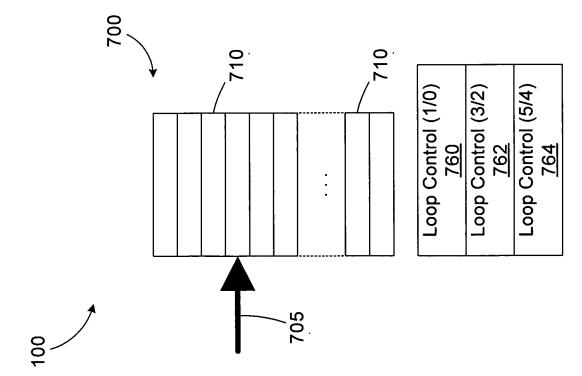




730
Init
do
Ľ

	Loop IP (0) 750	Loop IP (1) 751	Loop IP (2) 752	Loop IP (3) 753	Loop IP (4)  754	Loop IP (5) 755	Loop IP (6) 756
Repeat Counter 717	Loop Counter (0) 740	Loop Counter (1)	Loop Counter (2) 742	Loop Counter (3) 743	Loop Counter (4) 744	Loop Counter (5) 745	Loop Counter (6) 746
Repeat Count 715	Loop Count (0) 720	Loop Count (1) 721	Loop Count (2) 722	Loop Count (3) 723	Loop Count (4) 724	Loop Count (5) 725	Loop Count (6) 726

Figure 7



	9											9	1	9										OIP MIN	Thank out	•
BXCTL	616	4	4	4	9							-										-	<u> </u>	<u> </u>		
AXCTL	614	2	4	7	4	STORE0	630	11	1F	11	80		ATTN	644	0	0	0	0		marking start	/ one or loop 26)	962	store remaining number of iterations for 1st	er (740-746)		
ZMODE	612	0	0	0	0	ABYMODE	628	0	0	0	0		REPEAT	642	0	0	0	1		1st instruction.	Identify 1st instruction, marking start of 1st loop (defined by one or loop counts 720-726)		ining primber o	loop in a loop counter (740-746)		
ZPARM	610	0	0	0	0	ABXMODE	<u>626</u>	0	0	0	0		LOOP END	640	00	00	00	03	8	Identify		794	store rema	i dool	re 7A	
PARM3	809	0000	0000	0000	0000	ABYMUX	624	0	0	0	0		LOOP BEGIN	638	01	02	00	00	Figure 8			792	equent	ounter	Figure 7A	
PARM2	909	0000	01FF	3444	1000	ABXMUX	622	-	0	0	0		STORE3	<u>636</u>	1F	11	1F	1F			Store intructions in queue (700) indexed by instruction pointer (705)	798	Load instruction pointer (705) with a subsequent loop instruction pointer value if number of iterations	remaining for 1st loop is zero (e.g., loop counter [740-746] equals zero)		
PARM1	604	0000	0000	0000	0001	BYCTL	620	4	4	4	4		STORE2	634	1F	1F	1F	18				7	ction pointer (7)	or 1st loop is zero (e.g., 1740-746] equals zero)		
PARMO	<u>602</u>	07FF	0000	0003	FFFF	AYCTL	618	4	2	9	4		STORE1	632	1F	1F	1F	1F			Store intruc		Load instru	remaining f		
	200	802	,	803~	804七			•				L						<u></u> -	ı							